AD-A047 908 NORTHROP CORP HAWTHORNE CALIF ELECTRONICS DIV F/G 17/7
AN/BRN-7 COMPUTER PROGRAM SPECIFICATION. VOLUME III. SYNCHRONIZ--ETC(U)
OCT 73 N00039-73-C-0209
UNCLASSIFIED NORT-73-48-VOL-3 NL

1 OF 1
AD A047908

END
DATE
FILMED
1-78
DDC

AD-A047908

1

Northrop Corporation Electronics Division

NORTHROP

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

VOL III.

NORTHROP

Electronics Division

- [ ] This submittal applies to AN/BRN-7 (Submarine Ω) only.
- [ ] This submittal applies to AN/SRN-( ) (Hydrofoil Ω) only.
- [x] This submittal applies to both AN/BRN-7 and AN/SRN-( ).

CONTRACT NO: N00039-73-C-0209

PROGRAM NAME: AN/BRN-7

AD-A047908

CDRL No: A01D, A01E, A01F

Title of CDRL: Computer Program Design Specification
Computer Subprogram Design Document
Data Base Design Document

Title of Doc: AN/BRN-7 Computer Program Specification
NORT 73-48
Volume 2 thru 13

Date: 1/16/74

Initial Submittal: [x] | Release | APS 1-16-74

Resubmittal: [ ] | Authentication

Prepared by: System Integration Unit
Orgn. Name

A4734 X574
Orgn. No. and Ext.

Approved by:

Norman P. Clark — Supervisor of Preparing Orgn.

Bill Tyler — Program Mgr.

Proj. Engr./Proj. Mgr.

FORM 6512 (2-72)

NORTHROP

Electronics Division

Copy No. 2

NORT 73-48

AN/BRN-7 COMPUTER PROGRAM SPECIFICATION

Volume III

SYNCHRONIZATION SUBPROGRAM DESIGN

October 12, 1973

Approved by

G. W. Hauser
Director, Engineering
Navigation Department

Volume III
of the

AN/BRN-7 OMEGA COMPUTER
PROGRAM SPECIFICATION

CONTENTS

SECTION 1

SCOPE

1.1 IDENTIFICATION

Volume I, Submarine OMEGA Computer Program Performance Specification, defines the functional requirements for the Submarine OMEGA Computer Program which is used by the Submarine OMEGA Navigation System. The tape which defines the computer program is entitled AN/BRN-7 Navigation Program.

Volume II, Submarine OMEGA Computer Program Design Specification, allocates the functional requirements of Volume I to the computer routine and subprogram level.

This volume describes the subprogram designated as Synchronization, which has the abbreviation SY in the program listing (Volume XIII).

1.2 SUBPROGRAM TASKS

a) Set the antenna switching matrices to proper orientation for the collection of synchronization pulses.

b) Record the computer reference time.

c) Initiate the repetitive 10-second data collection process. Since this is a repetitive operation, alternate 10-second collections must be stored in separate storage areas.

d) After a 10-second data collection period the 100 correlation numbers must be calculated.

e) Test the four largest correlation values to find if synchronization can be determined.

f) Determine t_{sync}.

SECTION 2

APPLICABLE DOCUMENTS

a) Submarine OMEGA Computer Program Performance Specification (Volume I of the Submarine OMEGA Computer Program Specification).

Applicable Sections:

3.1 Introduction
3.3.1 Synchronization
3.3.2 Signal Input Timing
3.3.3 Antenna Switching Control

b) Submarine OMEGA Computer Program Design Specification (Volume II of the Submarine OMEGA Computer Program Specification). All sections.

c) NORT 71-41, NDC 1070, Macro Assembler, May 1971

d) NORT 68-115A, Detailed Description of NDC-1070 Computer Instructions, Revision A, February 1970

e) NORT 69-87A, NDC-1070 Flow Chart Program, User's Manual.

SECTION 3

REQUIREMENTS

In order to understand the subprogram description contained in the following pages, it is necessary that the reader will have become familiar with the associated functional requirements found in Volume I, Performance Specification, and with the subprogram allocation found in Volume II, Design Specification.

3.1 DETAILED DESCRIPTION

3.1.1 Reference Labels to Flow Diagrams

The code used to reference a particular data block in the flow diagrams, Section 3.2, is as follows: The first number is that page number found in the upper right corner of the diagrams. This will be followed by a slash sign (/) to separate the page number from the block designator. The designator will either be a mnemonic label (e.g., TEST SYNC), a local label indicated by a dollar sign ($), or an integer. The two types of labels reference the particular information block, on the given page, to which the label is attached. The integer number, n, means that the referenced block in the n^{th} block from the top of the page; p8/3 would refer to page 8 and the third information block down.

Finally, the label p1/$ 2 + 3 refers to page 1, and the 3rd information block after the label $ 2.

3.1.2 Executive Management of Synchronization Routines

The synchronization subprogram is defined by routines SYNC CALC (which uses the SYNC SUMMATION subroutine), SYNC1, SYNC2, and SYNC3. The manipulation of these routines is accomplished by both MAIN PROGRAM and the FIVE MILLISECOND INTERRUPT Routine (henceforth referred to as the INTERRUPT routine). To fully understand the sequencing of these synchronization routines, it is necessary to have an understanding of the interaction between MAIN PROGRAM and INTERRUPT, including the selection of Ω-tasks. This information is fully described in Volume II, Design Specification. A brief summary here will serve to review these relationships, and clarify the synchronization process as well as the flow diagrams.

MAIN PROGRAM has but one computational routine besides the MEMORY CHECKSUM Test; this is SYNC CALC with its subroutine SYNC SUMMATION. Before MAIN PROGRAM can perform the computations, the synchronization summation data must be collected and stored. This process is selected by INTERRUPT via SYNC1, SYNC2, and SYNC3.

At power-on, and after the software initialization process, MAIN PROGRAM will designate PREFLIGHT Test as an Ω-task, which is under the jurisdiction of INTERRUPT. PREFLIGHT Test sets up SYNC1 as the next Ω-task as well as the time interval at which it will be initiated by INTERRUPT. Upon completion of PREFLIGHT Test, sequencing returns to MAIN PROGRAM, which alternates between MEMORY CHECKSUM and the test of the 10-SEC PERIOD COMPLETE marker; essentially waiting for the INTERRUPT to take over control. It is the 10-sec. period complete marker which will be set by SYNC 3 to indicate that a complete set of data has been stored, and enable SYNC CALC to commence processing.

3.1.3 Description of Flow Diagrams

a) SYNC1 (Page 1)
Label: SYNC
P1/SYNC to $1

SYNC1 designates SYNC2 as the next Ω-task, and sets $\Delta t\Omega$ to 0.15 second (using OMEGA Subroutine) which specifies that 30 of the 5 ms interrupts will pass before entering SYNC2. SYNC1 then sets up the antenna switching matrix for the collection of synchronization data. Upon completion processing returns to MAIN PROGRAM, or to a Control-Indicator function if selected by the operator.

b) SYNC2 (Page 2)
Label: $2
p2/$2

SYNC2 will be entered at the 30th interrupt from initiation of SYNC1. SYNC2 will mark the computer reference time at this point. The determination of the time difference between this reference and the arrival of the first pulse from station A on 10.2 kHz is t_{sync}, and is the sole purpose of the synchronization process.

SYNC2 sets up SYNC 3 as the next Ω-task, (OMEGA Subroutine), sets $\Delta t\Omega = 0.1$ second, clears the six data input registers (although only two will be used per 10-second data period), illuminates the SYNC indicator on the Control-Indicator Panel, and sets the data base location for the first (of two) 200-word storage areas. These two storage areas will be alternated every 10 seconds so that one can fill while the other is being processed.

c) SYNC3 (Pages 3 to 5)
Label: $3
p3/$3

SYNC3 sets up itself on the next Ω-task, sets $\Delta t\Omega = 0.1$ second, then performs the Precision Frequency Generator (PFG) test.

p3/\$19,\$9

Restart process in case of PFG anomaly.

p3/\$8

There is a data counter (R,below) which will count from 0 to 198 by 2 in a 10-second period (a programming convenience). SYNC3 is entered every 0.1 second during synchronization, and the data counter will index by two counts each time.

There is also a frequency counter that tracks which of the three frequencies is being used, and, therefore,which two of the six input data words to store. This counter will be set at 0,2 or 4 respectively and sequentially every 10 seconds, and repetitively after passing 4.

p3/\$8 + 1

The sin/cos input words, representing the equation

$$X_j(R) = \sum_{i=1}^{20} \overline{\sin}\ \phi_{ij}$$

$$Y_j(R+1) = \sum_{i=1}^{20} \overline{\cos}\ \phi_{ij}$$

where j = 0,2,4 represents 10.2,13.6, or 11-1/3 kHz. X and Y are each composed of a summation of 20 DMA inputs from the correlators in the receiver strip which processes frequency j. The values X and Y will be associated with data count R and R + 1.

p4/2

The data counter, R, is incremented by 2 and tested. If R is less than 200 then R is saved(p5/\$4), and SYNC3 is exited, to return in another 0.1 second.

p4/\$5,\$6

If R equals the value 200, then data collection is complete for the 10-second period, the location and frequency of the collected data is saved for the SYNC CALC routine, and the alternate storage area is indicated for the next entrance to SYNC3 when another 10-second data collection period will begin.

p4/$6+2

The counter, N, will count the number of times that SYNC3 has completed a 10-second data collection period. N will be used in the correlation criterion later. N is incremented by 1.

p5/1

After completing each 10-second data collection period, SYNC3 will check to see if 64 data collection periods have passed since attempting synchronization.

If not, then N is saved and its inverse is computed with a scaling factor to be used in the processing of data.

If so, then N is set to the value 1 and saved. This will reinitialize synchronization.

The accumulated data from each 10-second period is added to that collected previously. This serves to average or smooth the data as time passes. After 640 seconds have elapsed with no synchronization, the decision is made here to start over under the assumption that synchronization cannot be completed with the existing data.

p5/$11+1

Set 10-SEC PERIOD COMPLETE marker. This will initiate the SYNC CALC routine. Exit SYNC3.

d) <u>SYNC CALC</u> (pages 6 through 11)
Label: SYNC CALC
(uses SYNC SUMMATION subroutine: pages 12 and 13)

When MAIN PROGRAM has been entered it has been alternately performing MEMORY CHECKSUM test and checking the 10-SEC PERIOD COMPLETE marker.

p6/SYNC CALC

SYNC has set the marker true. MAIN PROGRAM now sets it false, and proceeds to process the accumulated data.

p6/SYNC CALC+1

There are four "pointers" used to indicate the four largest values of the C(I); I=1,100. These four values will be referred to as C(1st), C(2nd), C(3rd), and C(4th). These are not to be confused with C(1), C(2), etc., which are the first few of the 100 correlation numbers in the set. Initialization sets C(100) to -0.5 and the pointers to the four largest are set to point to C(100) (a programming convenience);

sets up the indexing of the correlation numbers by 2's, and sets I=0 (first correlation index).

p6/$33 through p6/$3+2

Introduction: There are 100 sin/cos pairs in storage. Each pair will have an associated index value, R. For programming convenience these 100 pairs are stored individually as 0 to 199, and the associated index, R, runs from 0 to 198 by 2's. SYNC CALC will start with index R = I (which has been initialized to 0 above), compute a summation-dependent function (Figure 1) over all 200 items starting from R=0 and proceeding through R=198. The result will be C(0). Next the correlation index, I, will be 2, and the same summation will be taken starting from the data pair associated with storage index R=2 and running through R=198 and ending at R=0. This will be C(2). In this way 100 correlation values are obtained. The loop from $33 through $2 is that which calculates these 100 correlation numbers.

The relationship of the four largest C(I) values is used to decide whether synchronization can be established.

For the current index value I, initialize C(I)=0.

p6/$3

The assumption carried by each C(I) value on any frequency is that "Station A burst starts with the data pair associated with the current index I". As stated above, the synchronization summation determining the correlation number C for index I will start at R=I, run through 198, start at 0 and finish at R=I-2.

But this summation must not include station bursts which have been disallowed by the operator. For the present index, I, the program will calculate which sequences of the 100 data-pairs are not to be used in the summation due to their disallowal. However, on the 13.6 or 11-1/3 channel these sequences are not the same. It is the function represented by $3 which sets up a mathematical relationship (modulo 8) between the station number K, and frequency, to determine which data-pair sequences in the present collection represent each station.

p6/$3+1

Test the station, and therefore the data-pair sequence, for deselection. In order to explain this more fully, consider Figure 1 which represents the equations used. The equation for C(I) uses summations from R1 through R2 for the short slot sum, and R3 through R4 for the long slot sum. R1 through R4 are determined by a lower index KL_K and an upper index KU_K, which are fixed by the station integer K. Both KL_K and KU_K

$$\underset{(I=1,100)}{C(I)} = \sum_{K=1}^{8} S \left[\frac{\left(\sum_{R=R1}^{R2} X(R)\right)^2 + \left(\sum_{R=R1}^{R2} Y(R)\right)^2}{KU_K + 1 - KL_K} - \frac{\left(\sum_{R=R3}^{R4} X(R)\right)^2 + \left(\sum_{R=R3}^{R4} Y(R)\right)^2}{KU_K + 5 - KL_K} \right]$$

where

K	Slot Numbers counting from Correlation Index I: KL_K	KU_K
1	1	9
2	12	21
3	24	34
4	37	48
5	51	61
6	64	72
7	75	86
8	89	98

$$S = (10.2)\left[A(K=1) + B(K=2) + C(K=3) + D(K=4) + E(K=5) + F(K=6) + G(K=7) + H(K=8)\right]$$
$$+ (13.6)\left[A(K=2) + B(K=3) + C(K=4) + D(K=5) + E(K=6) + F(K=7) + G(K=8) + H(K=1)\right]$$
$$+ (11.3)\left[A(K=3) + B(K=4) + C(K=5) + D(K=6) + E(K=7) + F(K=8) + G(K=1) + H(K=2)\right]$$

$S = 1$ when logic state true
$S = 0$ when logic state false

and

$R1 = (I + KL_K) \bmod. 100$

$R2 = (KU_K + I) \bmod. 100$

$R3 = (I + KL_K - 2) \bmod. 100$

$R4 = (KU_K + I + 2) \bmod. 100$

FIGURE 1

determine slot numbers relative to the current index I. If $KL_K = 12$ and $I = 6$, then the storage location determined by KL_K is 18. Thus the station deselection variable, S, which is indicated as zero for station deselection, is mechanized in the computer program by not processing those data-pair sequences represented by K.

For example, suppose station B is deselected. In the equation for S the station value for B on 10.2 is K = 2; on 13.6, K = 3; and on 11-1/3, K = 4. Then, if SYNC CALC is processing data:

on 10.2 the summing sequence determined by $KL_K = 12$ and $KU_K = 21$ is not used;

on 13.6 the summing sequence determined by $KL_K = 24$ and $KU_K = 34$ is not used, and on

11-1/3 the summing sequence determined by $KL_K = 37$ and $KU_K = 48$ is not used.

The computer program uses the value K in the same way, except that it runs from 0 through 14 by 2 rather than 1 through 8 as indicated in the example. If the station is deselected, then the SYNC SUMMATION subroutine is bypassed and K is indexed at $2 and returns to the indexing relationship for the next station at $3.

p6/$3+2

Determines KL_K and KU_K relative to current index I.

p7/1

SYNC SUMMATION subroutine computes those squares of sums using running variables R1 and R2. This is the short slot sum, the first half of the C(I) equation within the bracket in Figure 1.

p7/2

Add the short slot sum to C(I).

p7/3

Compute R3 and R4.

p7/4

Call SYNC SUMMATION subroutine. Calculate the squares of sums, the long slot sum, using R3 and R4. This is the latter half of the bracketed equation for C(I).

p7/5

Subtract long slot sum from C(I).

p7/$2

Increment K to next station and return to $3 to complete C(I) calculations. If finished with the station-by-station summations, then continue.

p7/$2+1

C(I) is averaged with old correlation data for this same time slot 1.

$$C(I)_{ave.} = \frac{C(I) - C(I)_{last\ ave.}}{N} + C(I)_{last\ ave}$$

p7/$2+3 to p8/$4

Determine from all correlation numbers C(I) calculated so far (I from 0 to present value) which values of I for which C(I) is 1st, 2nd, and 3rd largest, and find the 4th largest value of C(I).

p8/$4

Index I by 2, and if I = 200, then SYNC CALC has accumulated one hundred C(I) values and continues. Otherwise, it returns sequencing to $33 to calculate the next C(I) value.

p9/1

The next two pages of flow chart require but brief explanation. SYNC CALC is trying to determine if synchronization is complete beyond a reasonable doubt. The reasonable doubt is the confidence criterion which is a function of C(1st) and N, the number of 10-second data collections:

$$C = \frac{0.576A^2 + 0.3C(1st)}{\sqrt{N}}$$

A = 240 pulses/0.1 sec

p9/3

If C(1st) is negative, then more data is needed and sequencing is returned to 4/TEST SYNC to await the completion of another 10-second data collection period. If C(1st) is positive, then continue.

p9/5

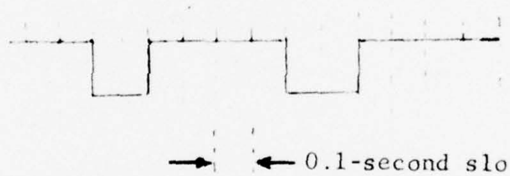
Test $|C(1st) - C(2nd)| > C$?

If so, then synchronization has been established and sequencing branches to p11/$10 and **establishes I associated with C(1st) as** *station A burst start time on* 10.2 kHz.

If not, then continue.

p9/6

Ideally the slots will be aligned such that there are always two 0.1-second sync slots between burst:



0.1-second slot

However, the start time for synchronization is arbitrary and it could be there is only one 0.1-second slot between bursts.

In this case, C(1st) and C(2nd) would be equal and adjacent. And in some cases the first three "best" correlation values could be adjacent. This case will also be tested later, if necessary.

p9/7

At this point, where C(1st) is not significantly greater than C(2nd), a test is made to determine if they are adjacent:

Is $|I(1st) - I(2nd)| = 2$? (remember that the indexing of I is by 2).

p9/8

Also, if the first slot and last slot, $I_a = 0$ and $I_b = 198$, are those slots such that $C(1st) = C(I_a)$ and $C(2nd) = C(I_b)$, then although they are adjacent in the cycle, they are not so in the program. This operation will determine if this is so and make the appropriate time adjustment. If both the above are false, then synchronization has failed and sequencing returns to Main Program to await another 10-second data period.

If either of the above is true, then C(1st) and C(2nd) together are best. Then in reality, C(3rd) is the next best and must be tested against C(1st) with the correlation criterion C as in p9/+5.

p10/$12+1

Test $|C(1st) - C(3rd)| > C$?

If so, then synchronization is complete and sequencing branches to p9/$13 where I(1st) and I(2nd) are averaged to determine the start time for station A burst on 10.2 kHz.

If the test fails, then it must be determined whether the intervals determined by C(1st) and C(3rd) are adjacent or not.

p10/$12+3

Is $|I(1st) - I(3rd)| = 2$?

p10/$12+4

Is $I(1st) = 0$ and $I(3rd) = 198$?

If both are false, then return to p4/TEST SYNC and await another data set. If either is true, then the first three best correlation values are nearly equal and C(4th) is moved to 2nd best and must be tested with C(1st) against the correlation criterion.

p10/$15+1

Is $C(1st) - C(4th) > C$?

If not, then return to p4/TEST SYNC and await the next data set. If true, then synchronization is complete and I(1st) determines the start of station A burst; i.e., it must be that I(2nd and I(3rd) were on either side of I(1st).

p11/\$16

By multiplying the selected index by 10 the program obtains the total number of 5-millisecond interrupts between the computer reference time recorded in SYNC3, and the start of station A burst on 10.2 kHz. This is t_{sync} compensation for the number of 10-second intervals consumed in synchronization will be done in the START OMEGA routine in the Executive Subprogram.

p11/\$11

The two storage areas are cleared for use by other portions of the OMEGA navigation programs, and the START OMEGA marker is set. This will halt SYNC3 from collecting data and return sequencing to the START OMEGA routine.

3.1.4 Description of Subroutines Used by the Synchronization Subprogram

The SYNC SUMMATION subroutine is the only subroutine not in the common category used by the Synchronization Subprogram. There are two arguments: a pointer and an interval.

Referring to Figure 1, there are four summations used in the calculation of each C(I). The limits are R1 through R2, R3 through R4, where R1 through R4 are dependent upon KL_K and KU_K.

p12/3

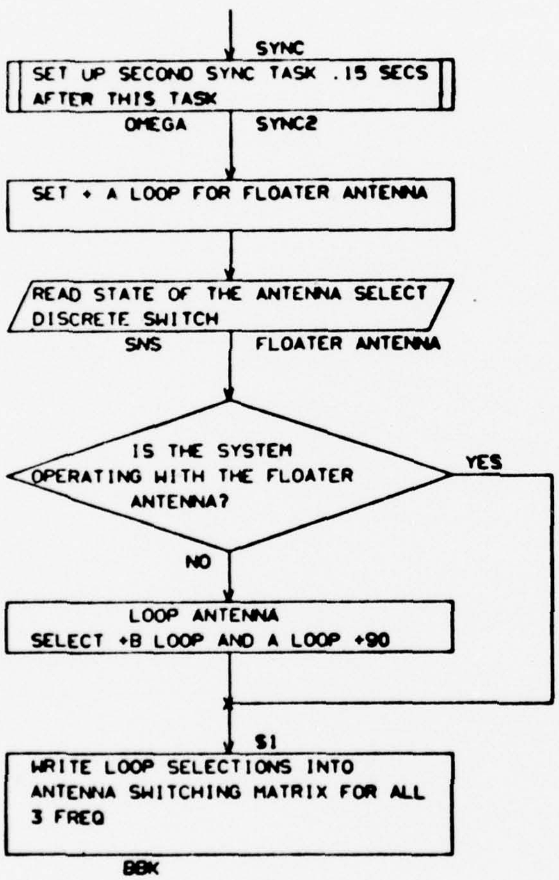
SYNC CALC sets the pointer at R3, and calculates the internal as R4-R3. The subroutine merely sums the input data values as indicated, and, upon completion, squares and sums, and then divides by the number of terms.

3.2 FLOW CHARTS

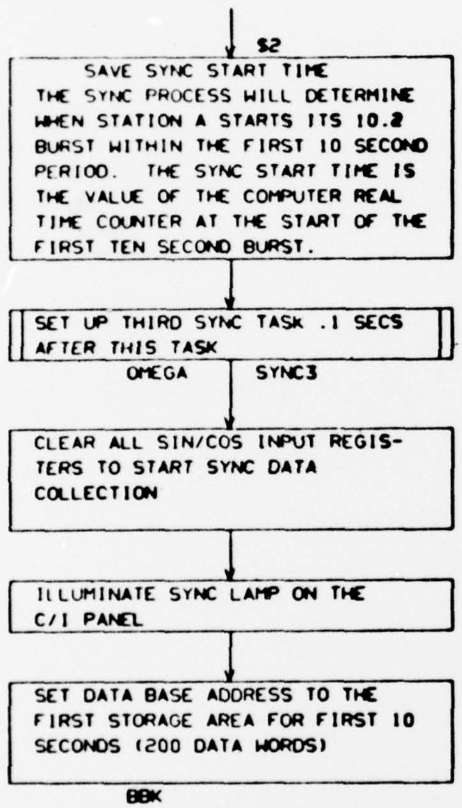
The Synchronization Subprogram flow charts are presented on the following pages.

PAGE 0001

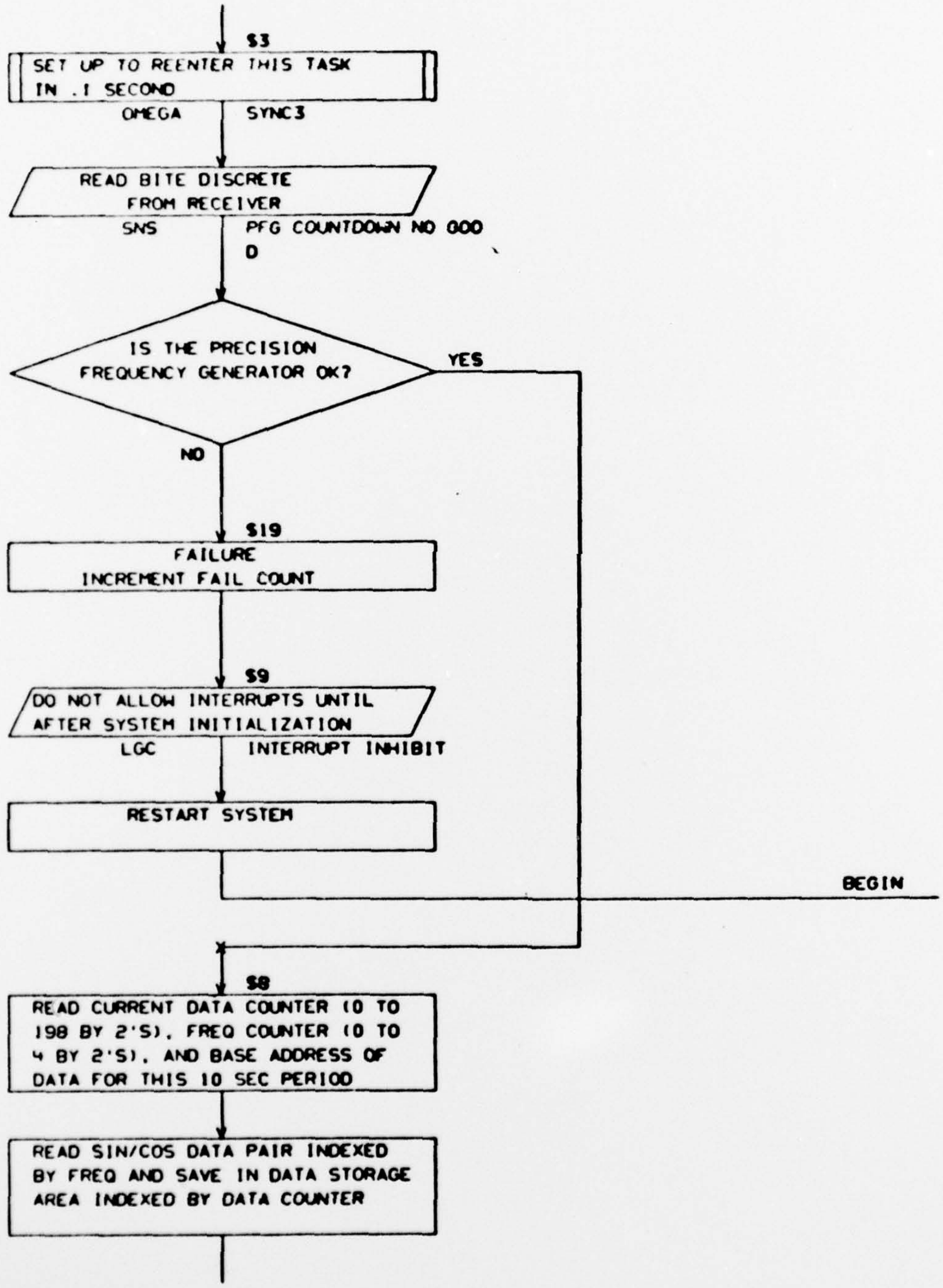
* SYNCHRONIZATION
*
* ENTERED .2 SEC AFTER PREFLIGHT TESTS



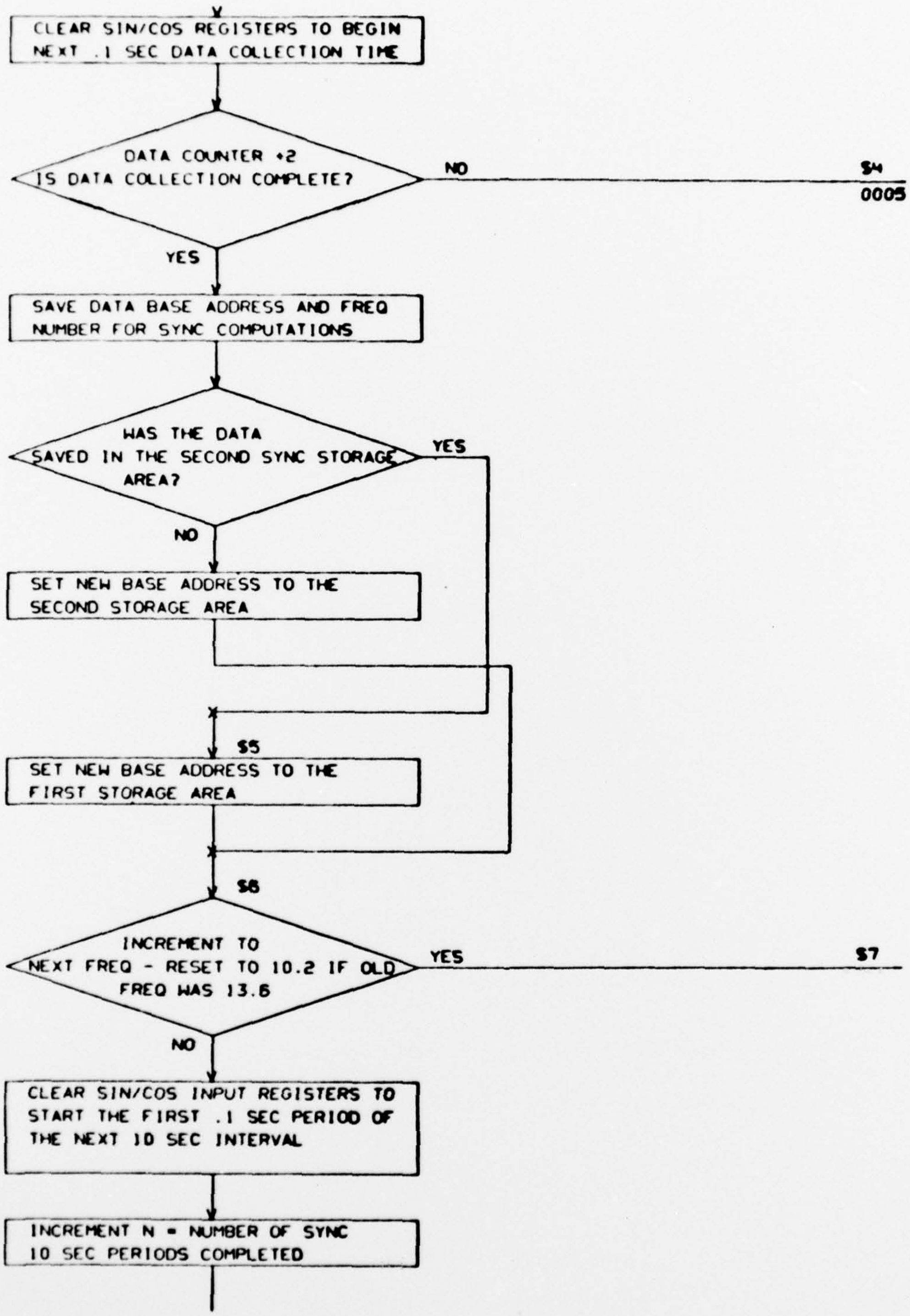
* SECOND SYNC TASK - CLEAR SIN/COS REGISTERS
* AND START FIRST SYNC 10 SECOND PERIOD

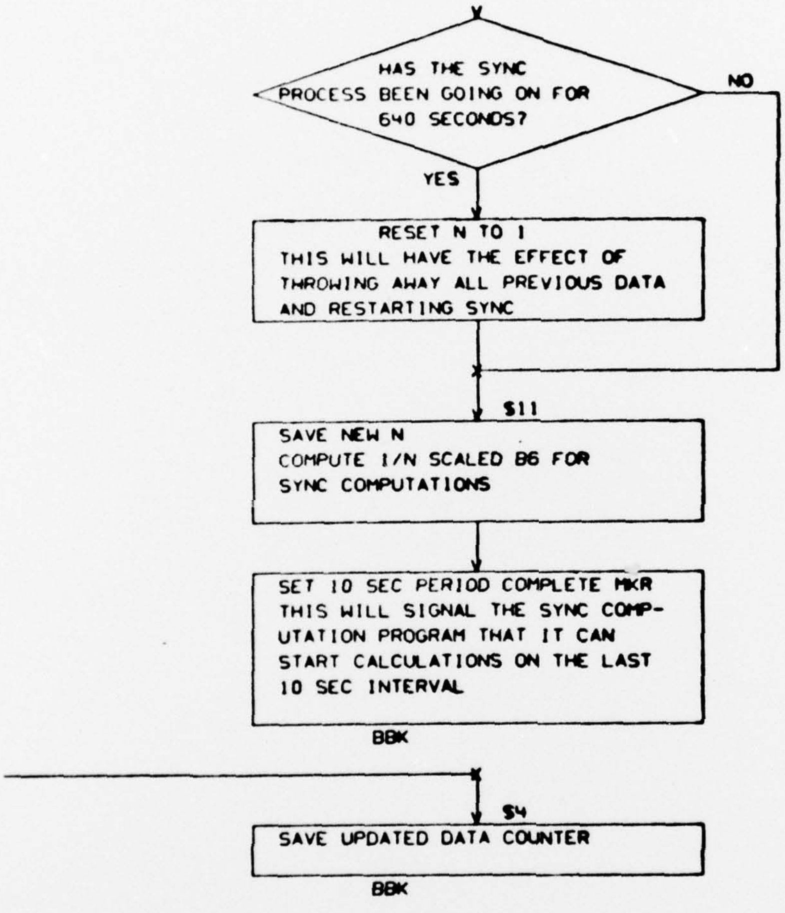


- THIRD SYNC TASK - DATA COLLECTION
- ENTERED EVERY .1 SEC UNTIL SYNC
- IS COMPLETE

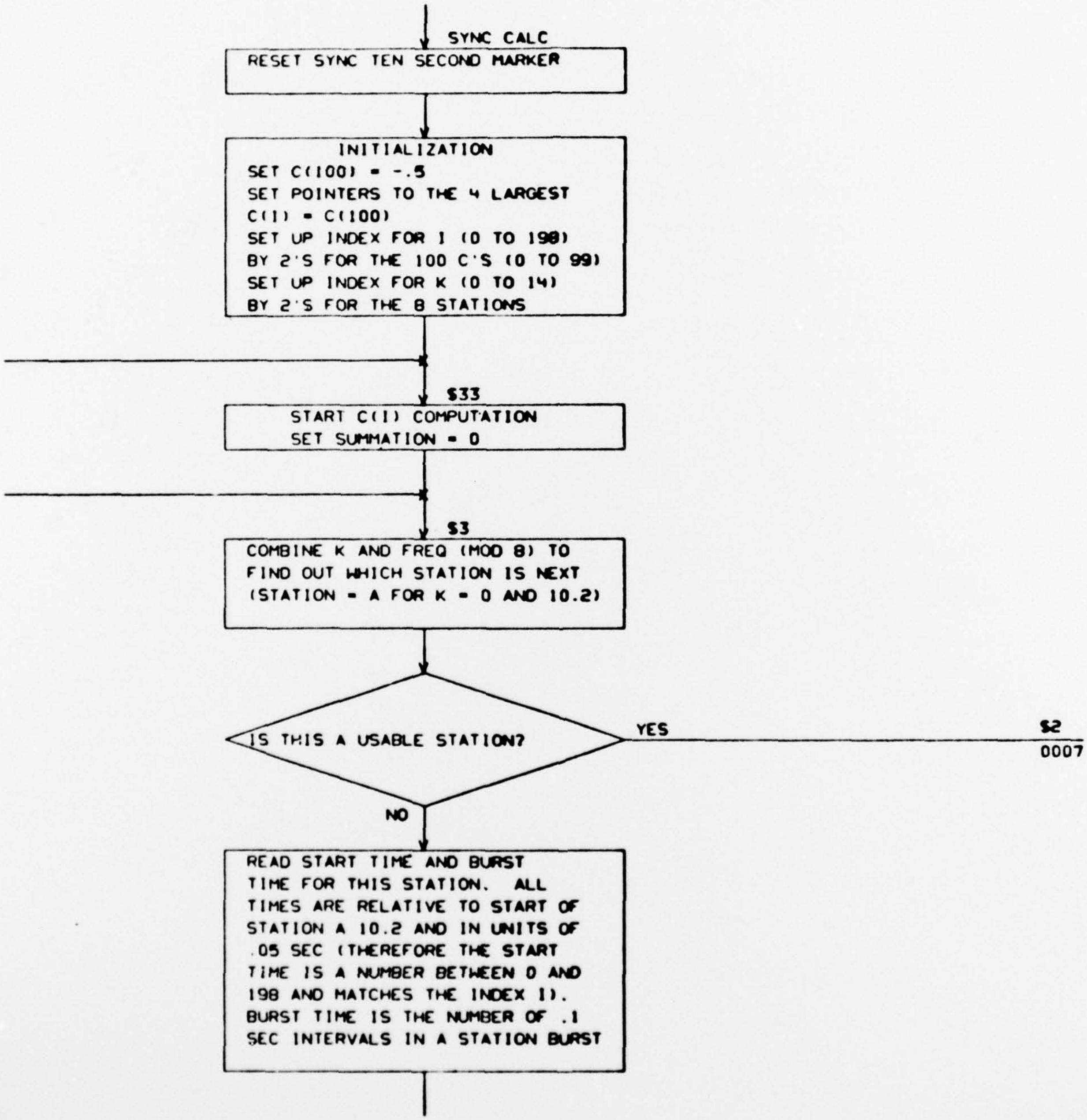


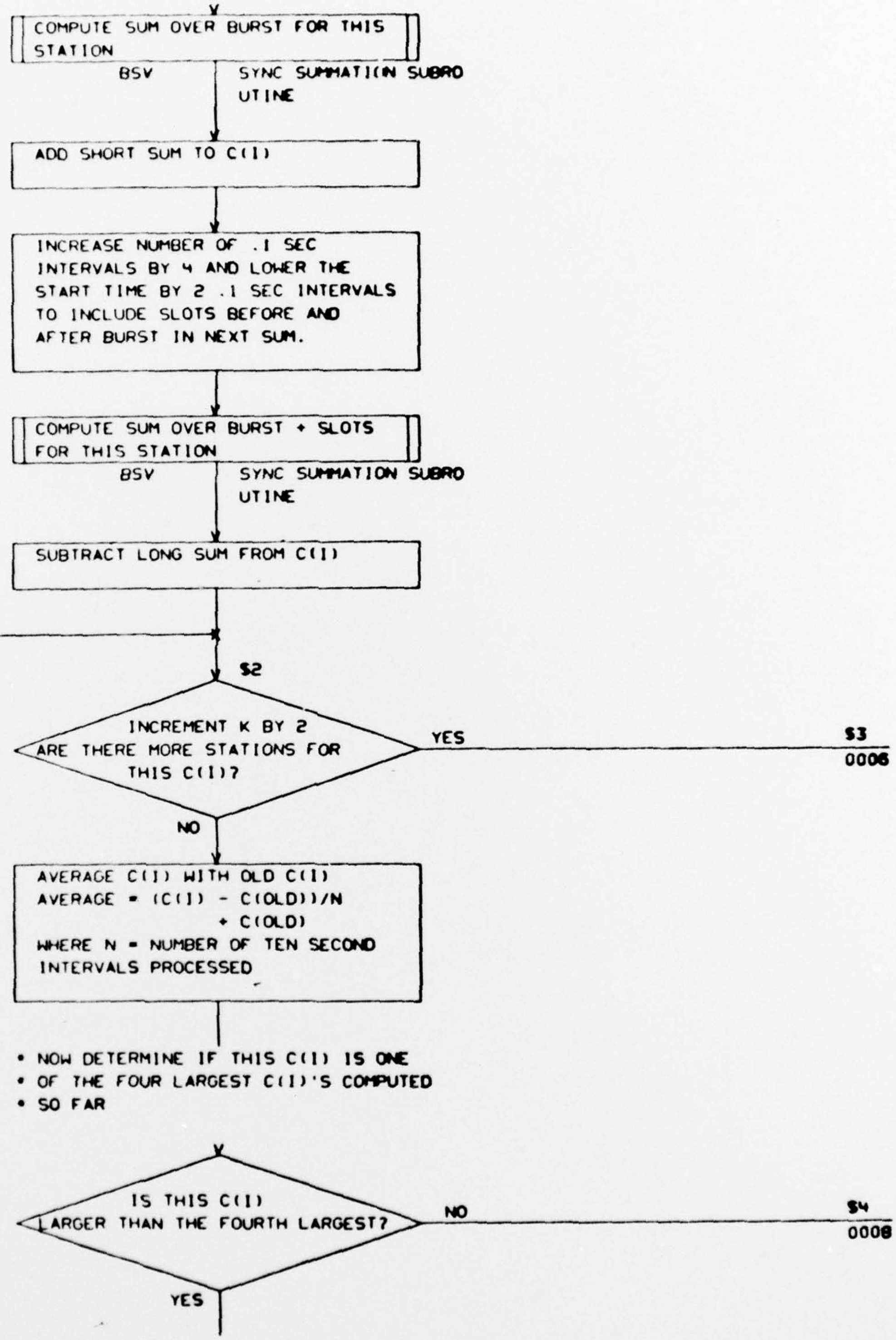
PAGE 0004

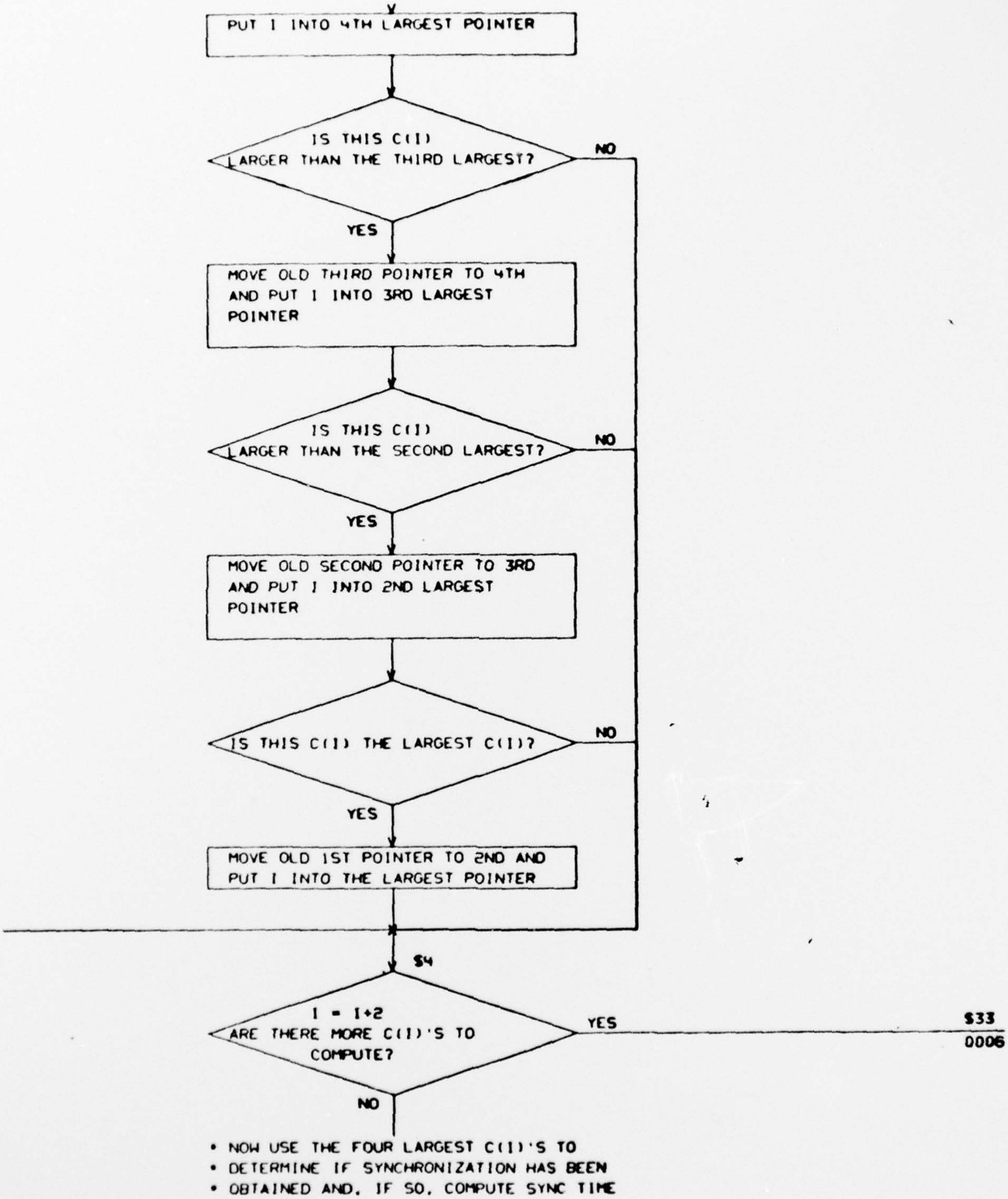

CLEAR SIN/COS REGISTERS TO BEGIN NEXT .1 SEC DATA COLLECTION TIME
DATA COUNTER +2 IS DATA COLLECTION COMPLETE?
NO
$4
0005
YES
SAVE DATA BASE ADDRESS AND FREQ NUMBER FOR SYNC COMPUTATIONS
HAS THE DATA SAVED IN THE SECOND SYNC STORAGE AREA?
YES
NO
SET NEW BASE ADDRESS TO THE SECOND STORAGE AREA
$5
SET NEW BASE ADDRESS TO THE FIRST STORAGE AREA
$6
INCREMENT TO NEXT FREQ - RESET TO 10.2 IF OLD FREQ WAS 13.6
YES
$7
NO
CLEAR SIN/COS INPUT REGISTERS TO START THE FIRST .1 SEC PERIOD OF THE NEXT 10 SEC INTERVAL
INCREMENT N = NUMBER OF SYNC 10 SEC PERIODS COMPLETED

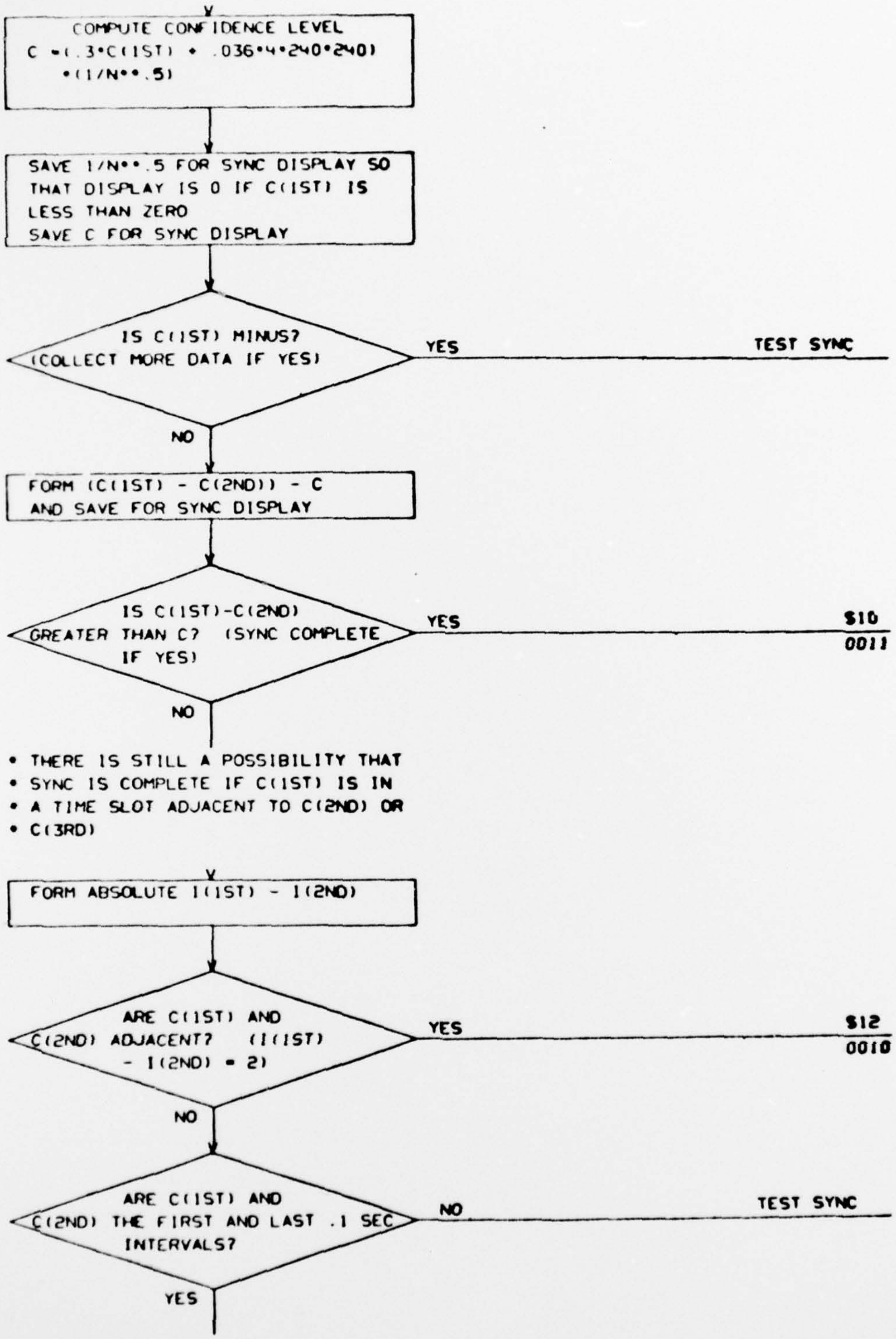

HAS THE SYNC PROCESS BEEN GOING ON FOR 640 SECONDS?
NO
YES
RESET N TO 1
THIS WILL HAVE THE EFFECT OF THROWING AWAY ALL PREVIOUS DATA AND RESTARTING SYNC
$11
SAVE NEW N
COMPUTE 1/N SCALED B6 FOR SYNC COMPUTATIONS
SET 10 SEC PERIOD COMPLETE MKR
THIS WILL SIGNAL THE SYNC COMPUTATION PROGRAM THAT IT CAN START CALCULATIONS ON THE LAST 10 SEC INTERVAL
BBK
$4
SAVE UPDATED DATA COUNTER
BBK

```
*                    SYNC CALCULATIONS
*
*  THIS PROGRAM IS ENTERED AFTER THE SYNC DATA COLLECTION ROUTINE
*  HAS COLLECTED DATA FOR TEN SECONDS.  IT WILL PROCESS THE DATA AND
*  DECIDE IF SYNC IS COMPLETE.
*
```

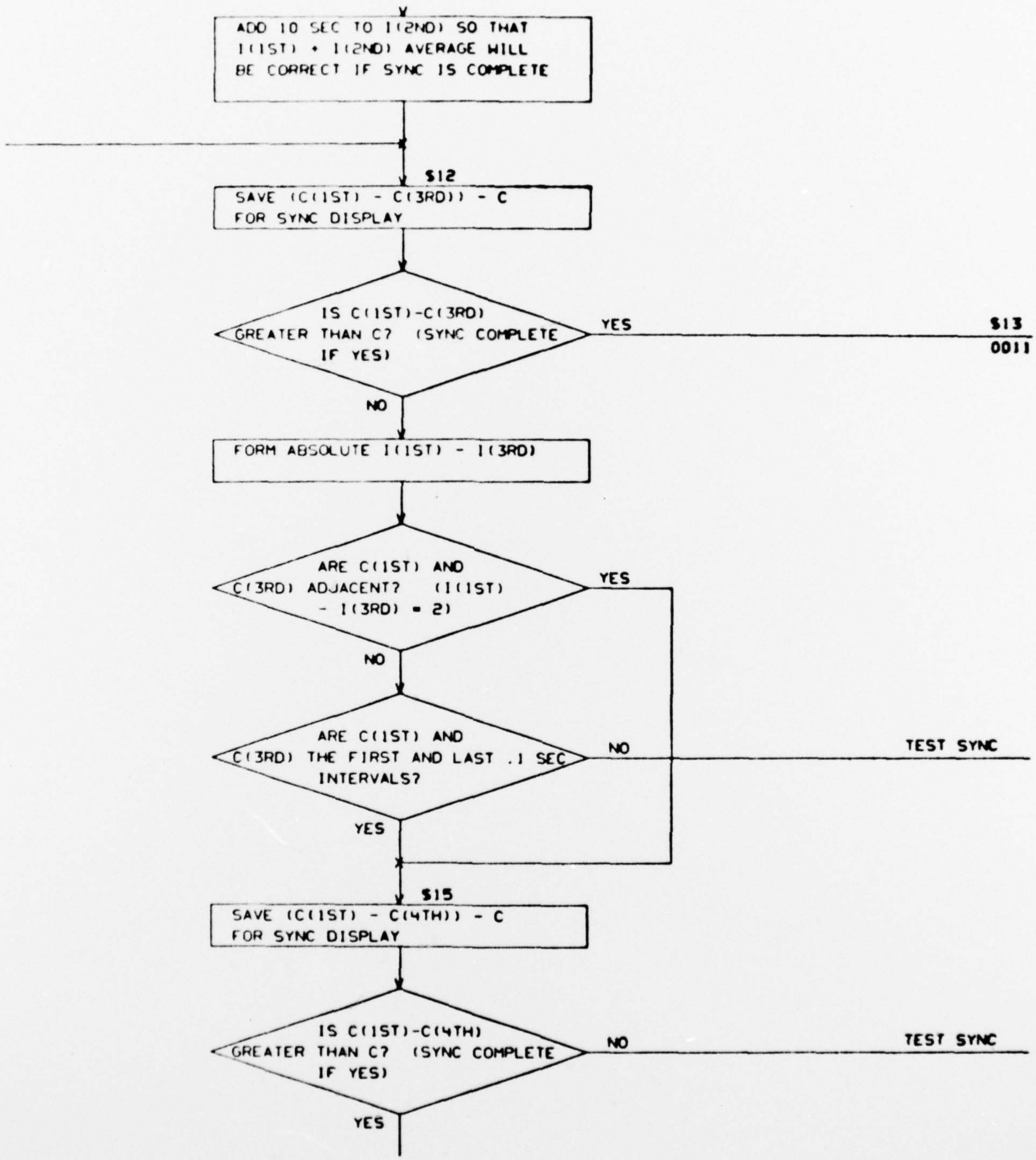


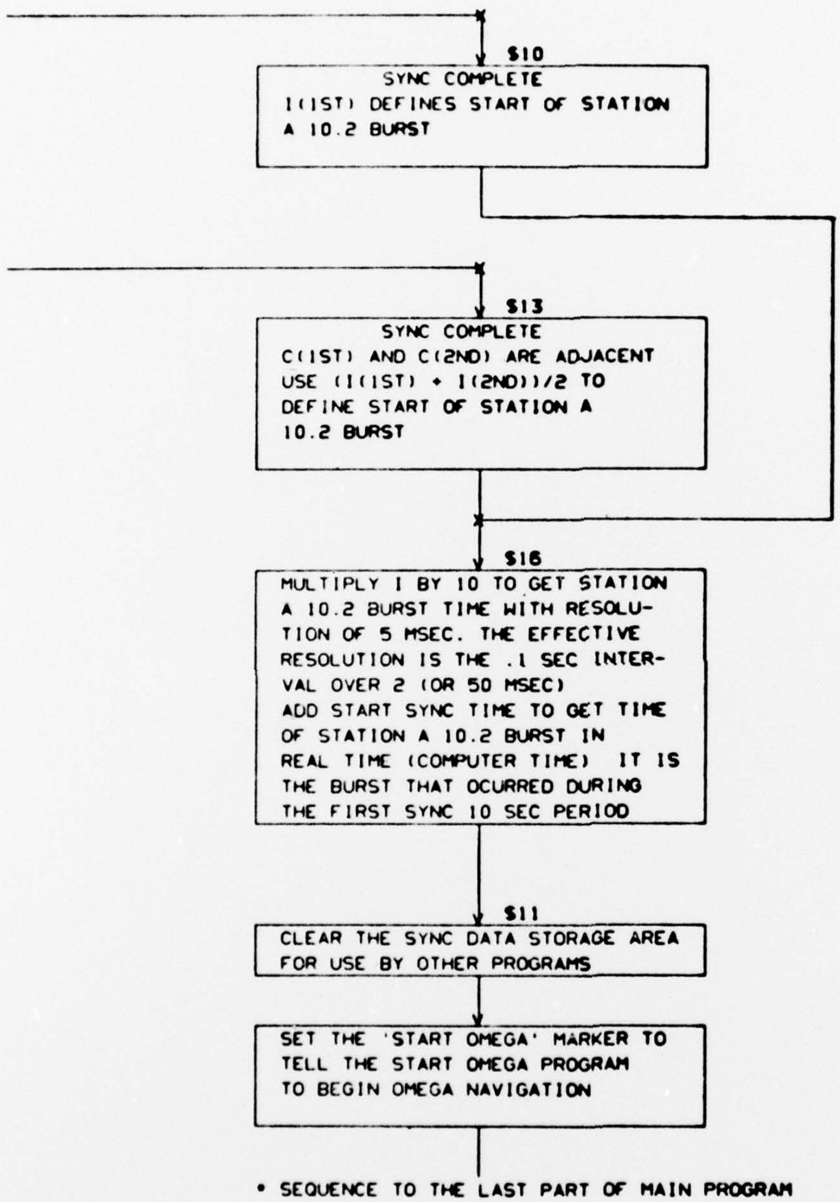

COMPUTE SUM OVER BURST FOR THIS STATION
BSV
SYNC SUMMATI(N SUBRO UTINE
ADD SHORT SUM TO C(I)
INCREASE NUMBER OF .1 SEC INTERVALS BY 4 AND LOWER THE START TIME BY 2 .1 SEC INTERVALS TO INCLUDE SLOTS BEFORE AND AFTER BURST IN NEXT SUM.
COMPUTE SUM OVER BURST + SLOTS FOR THIS STATION
BSV
SYNC SUMMATION SUBRO UTINE
SUBTRACT LONG SUM FROM C(I)
$2
INCREMENT K BY 2
ARE THERE MORE STATIONS FOR THIS C(I)?
YES
$3
0006
NO
AVERAGE C(I) WITH OLD C(I)
AVERAGE = (C(I) - C(OLD))/N + C(OLD)
WHERE N = NUMBER OF TEN SECOND INTERVALS PROCESSED
* NOW DETERMINE IF THIS C(I) IS ONE
* OF THE FOUR LARGEST C(I)'S COMPUTED
* SO FAR
IS THIS C(I) LARGER THAN THE FOURTH LARGEST?
NO
$4
0008
YES


PUT I INTO 4TH LARGEST POINTER
IS THIS C(I) LARGER THAN THE THIRD LARGEST?
NO
YES
MOVE OLD THIRD POINTER TO 4TH AND PUT I INTO 3RD LARGEST POINTER
IS THIS C(I) LARGER THAN THE SECOND LARGEST?
NO
YES
MOVE OLD SECOND POINTER TO 3RD AND PUT I INTO 2ND LARGEST POINTER
IS THIS C(I) THE LARGEST C(I)?
NO
YES
MOVE OLD 1ST POINTER TO 2ND AND PUT I INTO THE LARGEST POINTER
$4
I = I+2 ARE THERE MORE C(I)'S TO COMPUTE?
YES
$33
0006
NO
• NOW USE THE FOUR LARGEST C(I)'S TO
• DETERMINE IF SYNCHRONIZATION HAS BEEN
• OBTAINED AND, IF SO, COMPUTE SYNC TIME

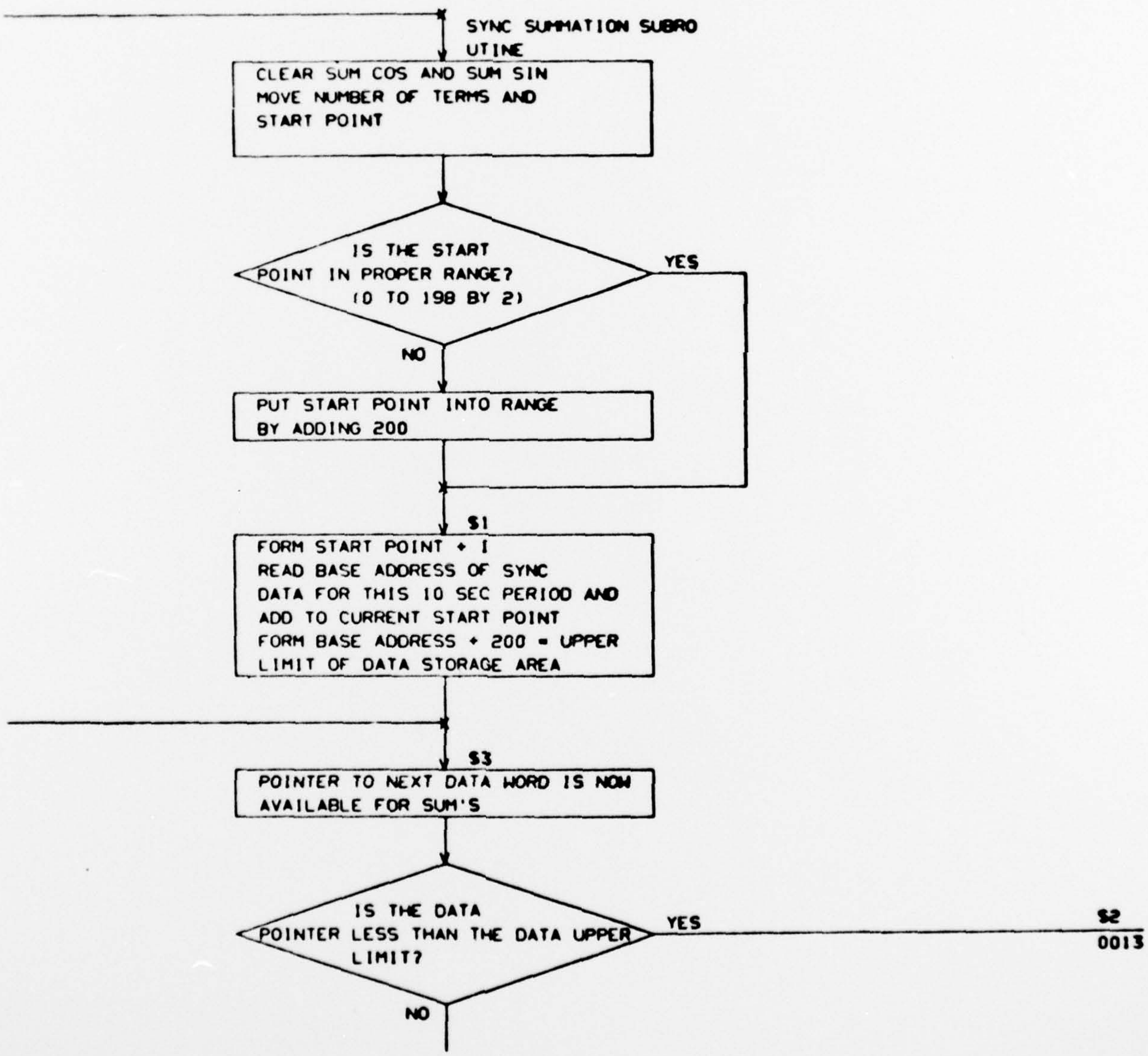


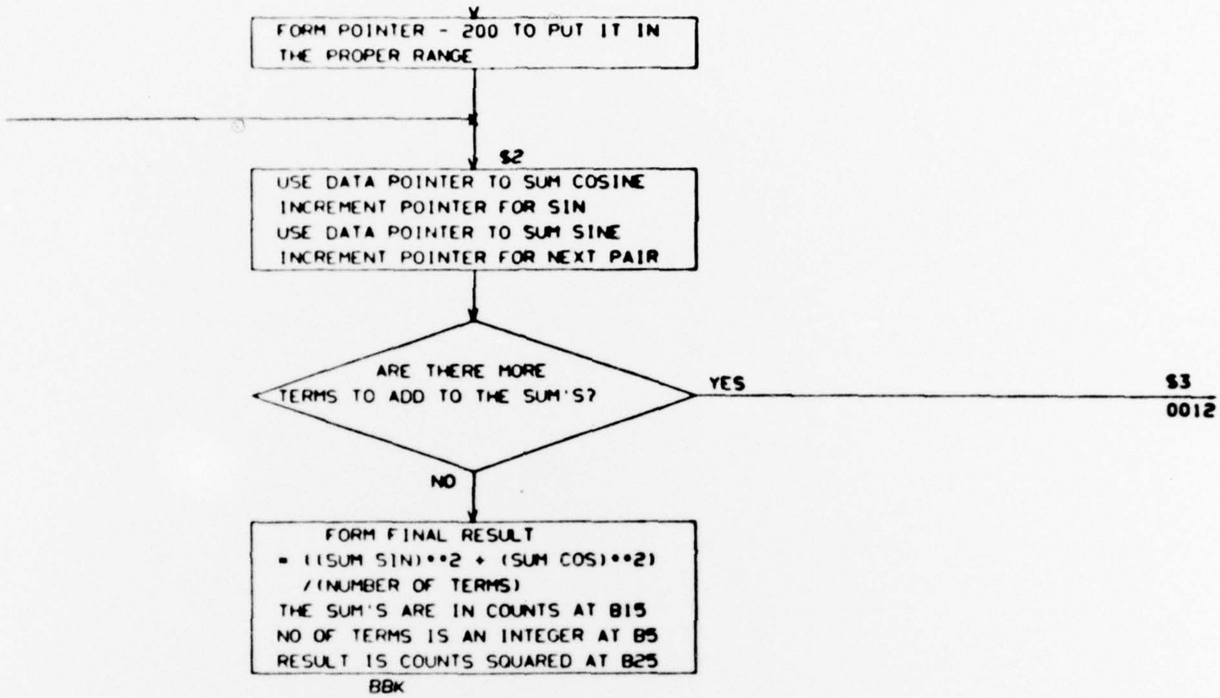
COMPUTE CONFIDENCE LEVEL
C =(.3*C(1ST) + .036*4*240*240)
*(1/N**.5)
SAVE 1/N**.5 FOR SYNC DISPLAY SO
THAT DISPLAY IS 0 IF C(1ST) IS
LESS THAN ZERO
SAVE C FOR SYNC DISPLAY
IS C(1ST) MINUS?
(COLLECT MORE DATA IF YES)
YES
TEST SYNC
NO
FORM (C(1ST) - C(2ND)) - C
AND SAVE FOR SYNC DISPLAY
IS C(1ST)-C(2ND)
GREATER THAN C? (SYNC COMPLETE
IF YES)
YES
$10
0011
NO
* THERE IS STILL A POSSIBILITY THAT
* SYNC IS COMPLETE IF C(1ST) IS IN
* A TIME SLOT ADJACENT TO C(2ND) OR
* C(3RD)
FORM ABSOLUTE I(1ST) - I(2ND)
ARE C(1ST) AND
C(2ND) ADJACENT? (I(1ST)
- I(2ND) = 2)
YES
$12
0010
NO
ARE C(1ST) AND
C(2ND) THE FIRST AND LAST .1 SEC
INTERVALS?
NO
TEST SYNC
YES


ADD 10 SEC TO I(2ND) SO THAT I(1ST) + I(2ND) AVERAGE WILL BE CORRECT IF SYNC IS COMPLETE
$12
SAVE (C(1ST) - C(3RD)) - C FOR SYNC DISPLAY
IS C(1ST)-C(3RD) GREATER THAN C? (SYNC COMPLETE IF YES)
YES
$13
0011
NO
FORM ABSOLUTE I(1ST) - I(3RD)
ARE C(1ST) AND C(3RD) ADJACENT? (I(1ST) - I(3RD) = 2)
YES
NO
ARE C(1ST) AND C(3RD) THE FIRST AND LAST .1 SEC INTERVALS?
NO
TEST SYNC
YES
$15
SAVE (C(1ST) - C(4TH)) - C FOR SYNC DISPLAY
IS C(1ST)-C(4TH) GREATER THAN C? (SYNC COMPLETE IF YES)
NO
TEST SYNC
YES


$10
SYNC COMPLETE
I(1ST) DEFINES START OF STATION A 10.2 BURST
$13
SYNC COMPLETE
C(1ST) AND C(2ND) ARE ADJACENT USE (I(1ST) + I(2ND))/2 TO DEFINE START OF STATION A 10.2 BURST
$16
MULTIPLY I BY 10 TO GET STATION A 10.2 BURST TIME WITH RESOLUTION OF 5 MSEC. THE EFFECTIVE RESOLUTION IS THE .1 SEC INTERVAL OVER 2 (OR 50 MSEC)
ADD START SYNC TIME TO GET TIME OF STATION A 10.2 BURST IN REAL TIME (COMPUTER TIME) IT IS THE BURST THAT OCURRED DURING THE FIRST SYNC 10 SEC PERIOD
$11
CLEAR THE SYNC DATA STORAGE AREA FOR USE BY OTHER PROGRAMS
SET THE 'START OMEGA' MARKER TO TELL THE START OMEGA PROGRAM TO BEGIN OMEGA NAVIGATION
• SEQUENCE TO THE LAST PART OF MAIN PROGRAM

```
* SYNC SUMMATION ROUTINE
*
* THIS ROUTINE IS USED BY THE SYNC COMPUTATIONAL
* PROGRAM TO FORM THE SUMMATION OF THE SINE'S
* AND COSINE'S OVER THE INTERVAL SPECIFIED BY
* THE ARGUMENTS.  THE ARGUMENTS CONSIST OF THE
* NUMBER OF TERMS, NUMBER OF THE FIRST TERM
* (ASSUMING I=0) AND THE CURRENT VALUE OF I. THE
* ARGUMENTS ARE NOT DISTURBED AND THE RESULT IS
* LEFT IN REGISTERS ADDED BY THIS ROUTINE.
*
```




FORM POINTER - 200 TO PUT IT IN
THE PROPER RANGE
$2
USE DATA POINTER TO SUM COSINE
INCREMENT POINTER FOR SIN
USE DATA POINTER TO SUM SINE
INCREMENT POINTER FOR NEXT PAIR
ARE THERE MORE
TERMS TO ADD TO THE SUM'S?
YES
$3
0012
NO
FORM FINAL RESULT
= ((SUM SIN)**2 + (SUM COS)**2)
/(NUMBER OF TERMS)
THE SUM'S ARE IN COUNTS AT B15
NO OF TERMS IS AN INTEGER AT B5
RESULT IS COUNTS SQUARED AT B25
BBK

3.3 COMPUTER SUBPROGRAM ENVIRONMENT

3.3.1 SYNC LIMIT TABLE

The purpose of the SYNC LIMIT TABLE is to define the station pattern for the SYNC CALC routine. The table has eight entries that define the start point of each station in the burst and the length of burst.

The 10-second station pattern is divided into 200 intervals to match the I counter used in SYNC CALC. Station A is assumed to start sending 10.2 during the first interval (0). Each entry contains two words with the first word identifying the start point of the station burst and the second containing the number of intervals that the station transmits. The eight entries are ordered by station. The label of this table is SYNC LIMIT TABLE.

3.3.2 SYNC DATA Storage

The synchronization process requires over 600 words of data storage for the two raw data storage areas, C(I)'s and miscellaneous markers and counters. Because the other OMEGA functions cannot begin until the synchronization process is complete, the 600 words are shared between synchronization and the other OMEGA functions. Only parameters not used during synchronization may be placed in the sync data storage area by the other OMEGA programs. See SYNC DATA in the listing for the exact limits of the overlay area.

3.3.3 Input/Output Formats

DMA inputs 11_{16} through 16_{16} (see Table 3.2-2, Volume I) receive the six phase counter inputs from the receiver strips:

- 11_{16} and 12_{16} receive cosine and sine respectively for 10.2 kHz,
- 13_{16} and 14_{16} receive cosine and sine respectively for 13.6 kHz, and
- 15_{16} and 16_{16} receive cosine and sine respectively for 11-1/3 kHz.

These words are updated every 5 milliseconds with the update occurring during the last one-third of the 5-millisecond period that is initiated by the time signal from the receiver. The update consists of reading a word from DMA, adding the output of the phase counter and writing the new word back into DMA. This requires approximately 181 microseconds from the read to the write operation. The LSB for all updating is bit one.

3.3.4 Required System Library Subroutines

Subroutine	Functional Description (Section 3.1.3)	Flow Diagram	Subprogram Design Document (by Volume Number)
1) OMEGA	a) SYNC1 b) SYNC2 c) SYNC3	p1/SYNC p2/\$2+1 p3/\$3	IX Executive
2) SQRT	d) SYNC CALC	p9/1	XII Common
3) SYNC SUMMATION	d) SYNC CALC	p17/1 p17/4	III Synchronization